Supplementary Information

In Silico High Throughput Screening of Bimetallic and Single Atom Alloys using Machine Learning and Ab Initio Microkinetic Modelling

Shivam Saxena[1,$], Tuhin S. Khan[1,$,*], Fatima Jalid[1,3], Manojkumar Ramteke[2], M. Ali Haider[1,*]

1. Renewable Energy and Chemicals Lab, Department of Chemical Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, India
2. Department of Chemical Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, India
3. Department of Chemical Engineering, National Institute of Technology Srinagar, Srinagar, Jammu and Kashmir, India

$ equal contribution
* corresponding authors email address: tuhinsk@iitd.com, haider@iitd.ac.in

Table SI-1. Train and Test errors with different feature set to remove the least important features from the model.

No. of Features in Model	Train Error	Test Error
Top 27	0.003	0.31
Top 25	0.003	0.32
Top 20	0.003	0.33
Top 15	0.003	0.33

Table SI-2. Effect on training and testing Error with change of test/train data ratio for GBR model for predicting binding energy of carbon on A_3B bimetallic alloy (211 AA terminated surface)

Test/Train Split	Train Error	Test Error
15%/85%	0.0003	0.34
20%/80%	0.0003	0.34
25%/75%	0.0003	0.36
30%/70%	0.0003	0.36
50%/50%	0.0003	0.41

Table SI-3. Effect on training and testing Error with change of test/train data ratio for GBR model for predicting binding energy of oxygen on A_3B bimetallic alloy (211 AB terminated surface)

Test/Train Split	Train Error	Test Error
15%/85%	0.0003	0.37
20%/80%	0.0003	0.38
25%/75%	0.0003	0.40
30%/70%	0.0003	0.41
50%/50%	0.0003	0.46

Table SI-4. Effect on training and testing Error with change of test/train data ratio for GBR model for predicting binding energy of carbon on A_3B bimetallic alloy (211 AB terminated surface)

Test/Train Split	Train Error	Test Error
15%/85%	0.0003	0.35
20%/80%	0.0003	0.35
25%/75%	0.0003	0.36
30%/70%	0.0003	0.36
50%/50%	0.0003	0.39

Table SI-5. Effect on training and testing Error with change of test/train data ratio for GBR model for predicting binding energy of oxygen on SAA bimetallic (111) surface

Test/Train Split	Train Error	Test Error
15%/85%	0.0003	0.38
20%/80%	0.0003	0.36
25%/75%	0.0003	0.49
30%/70%	0.0003	0.51
50%/50%	0.0003	0.62

Table SI-6. Effect on training and testing Error with change of test/train data ratio for GBR model for predicting binding energy of carbon on SAA bimetallic (111) surface

Test/Train Split	Train Error	Test Error
15%/85%	0.0003	0.38
20%/80%	0.0003	0.37
25%/75%	0.0003	0.40

30%/70%	0.0003	0.44
50%/50%	0.0003	0.54

Binding configuration of oxygen and carbon atom on Cu-based SAAs.

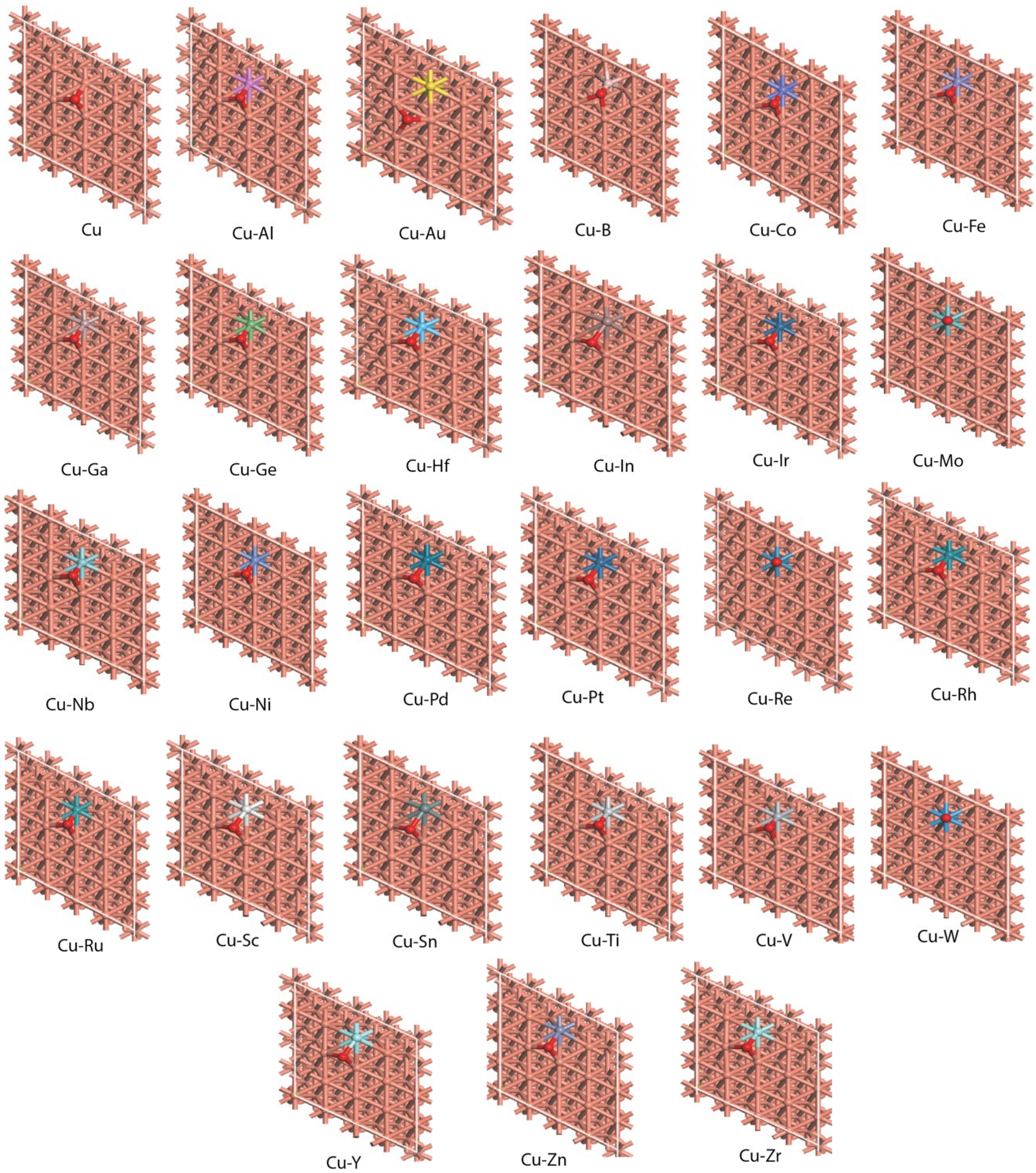


Figure SI-1. Binding configuration of oxygen over Cu-based SAAs. The adsorbate oxygen is shown in red color. The Cu atoms (matrix) are shown in coral color, whereas the single alloy atom has been shown in different colors, Al (purple) , Au (yellow), B (whitish purple), Co (blue), Fe (violet), Ga (gray-brown), Ge (green), Hf (sky bule), In (brown), Ir (dark blue), Mo (greenish sky blue), Nb (sky blue), Ni (violet-blue), Pd (greenish dark blue), Pt (navy blue), Re (blue), Rh (greenish blue), Ru (blue), Sc (shite), Sn (grey), Ti (off white), V (whitish grey), W (whitish blue), Y (sky blue), Zn (grey-violet) Zr (whitish sky blue).

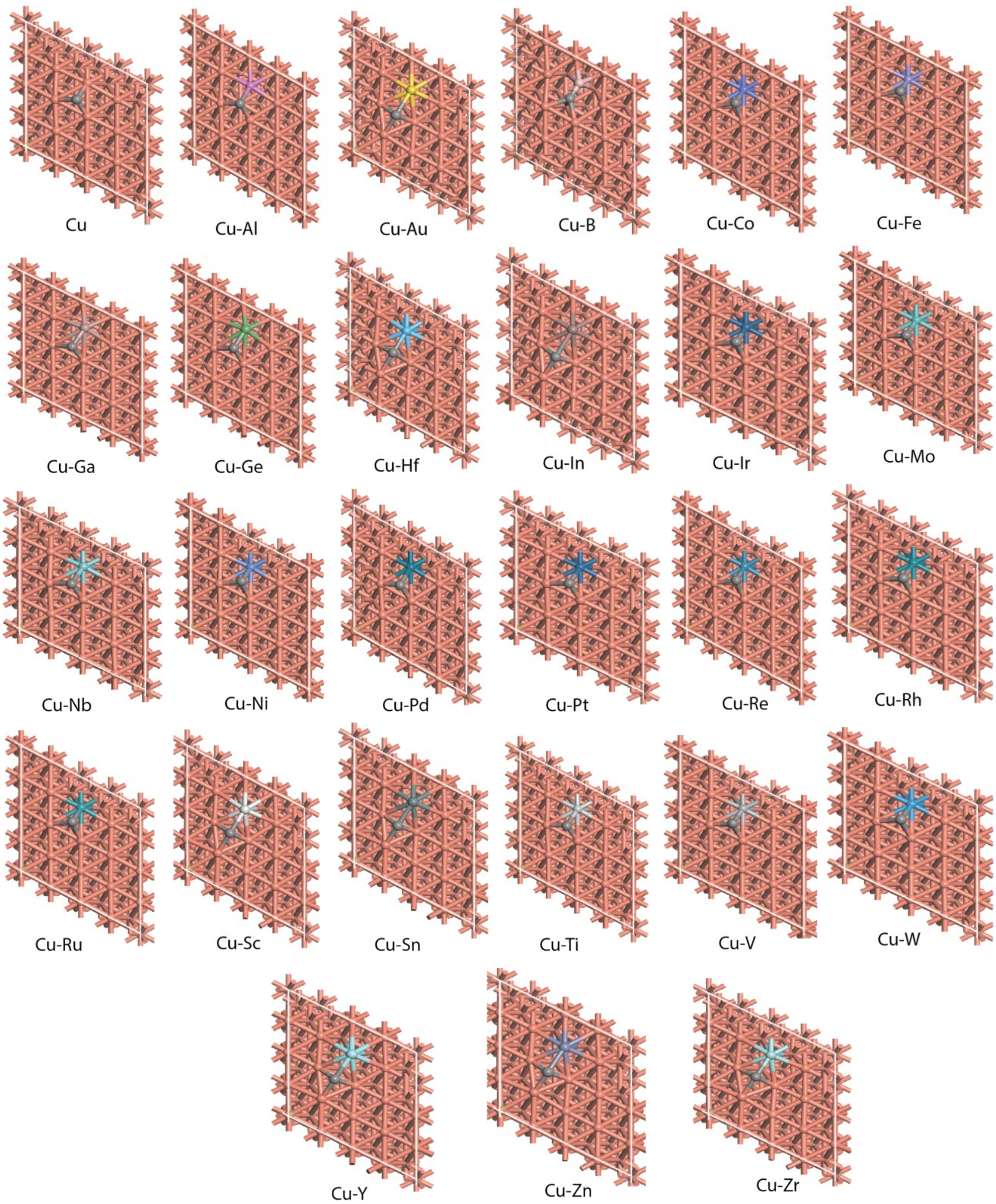


Figure SI-2. Binding configuration of carbon over Cu-based SAAs. The adsorbate carbon is shown in black color. The Cu atoms (matrix) are shown in coral color, whereas the single alloy atom has been shown in different colors, Al (purple) , Au (yellow), B (whitish purple), Co (blue), Fe (violet), Ga (gray-brown), Ge (green), Hf (sky bule), In (brown), Ir (dark blue), Mo (greenish sky blue), Nb (sky blue), Ni (violet-blue), Pd (greenish dark blue), Pt (navy blue), Re (blue), Rh (greenish blue), Ru (blue), Sc (shite), Sn (grey), Ti (off white), V (whitish grey), W (whitish blue), Y (sky blue), Zn (grey-violet) Zr (whitish sky blue).

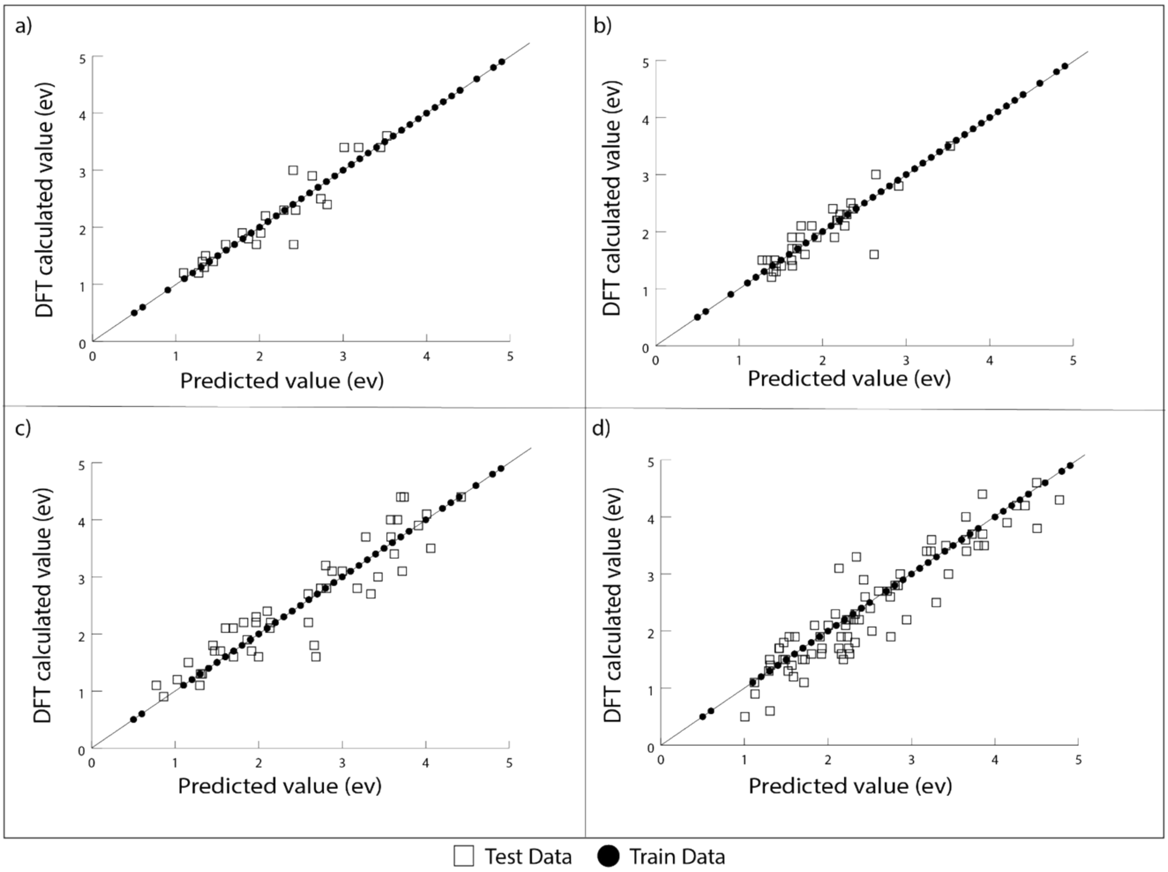


Figure SI-3. The deviation of DFT calculated carbon binding energy with that predicted from the GBR model for AA terminated A_3B bimetallic alloy for a) test/train ratio of 15/85 b) test/train ratio of 20/80 c) test/train ratio of 30/70 d) test/train ratio of 50/50

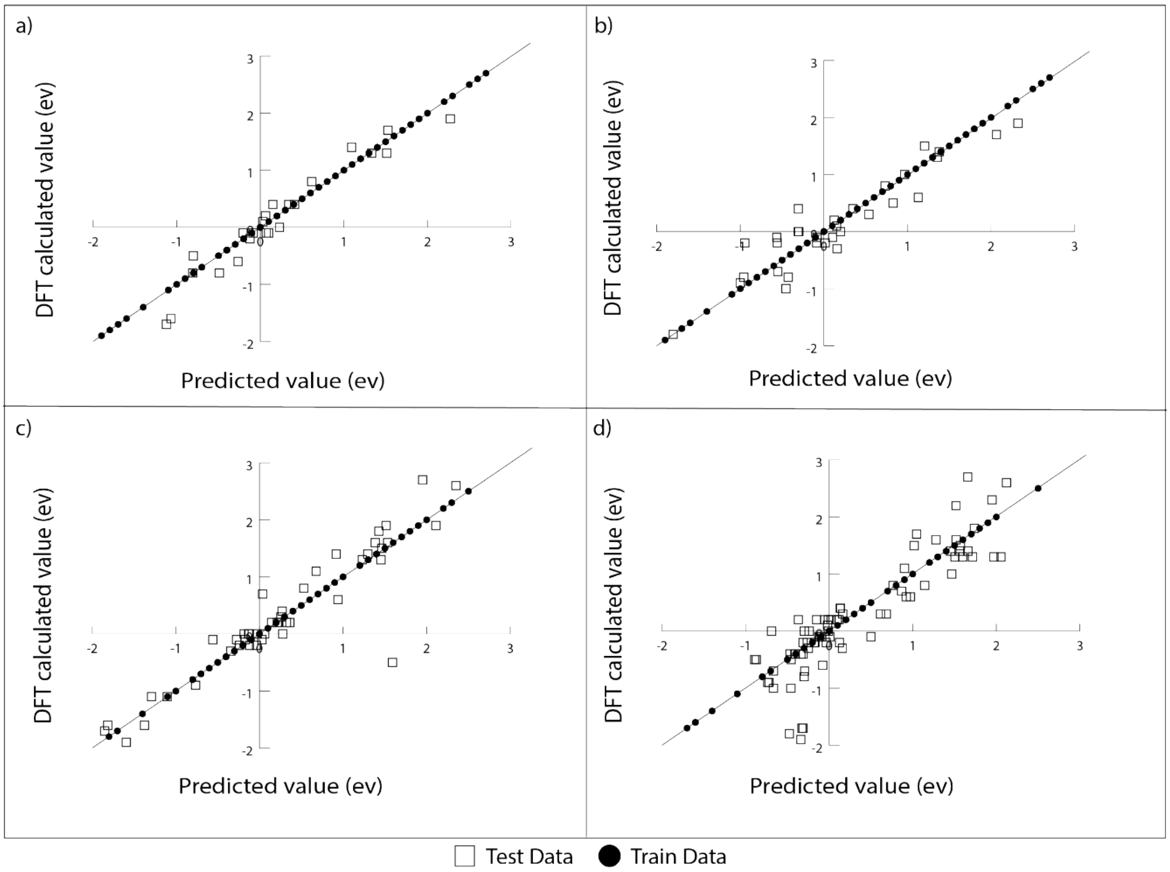


Figure SI-4. The deviation of DFT calculated oxygen binding energy with that predicted from the GBR model for AB terminated A_3B bimetallic alloy for a) test/train ratio of 15/85 b) test/train ratio of 20/80 c) test/train ratio of 30/70 d) test/train ratio of 50/50

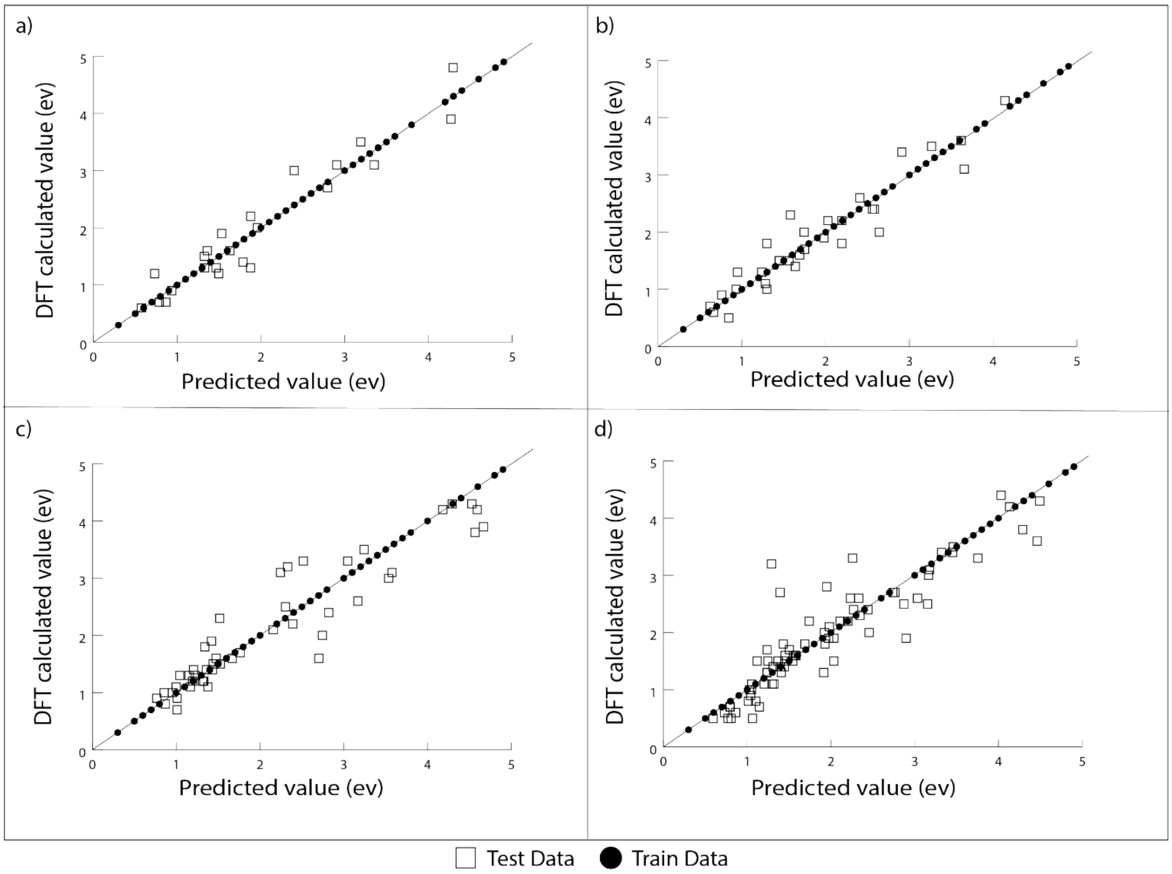


Figure SI-5. The deviation of DFT calculated carbon binding energy with that predicted from the GBR model for AB terminated A_3B bimetallic alloy for a) test/train ratio of 15/85 b) test/train ratio of 20/80 c) test/train ratio of 30/70 d) test/train ratio of 50/50

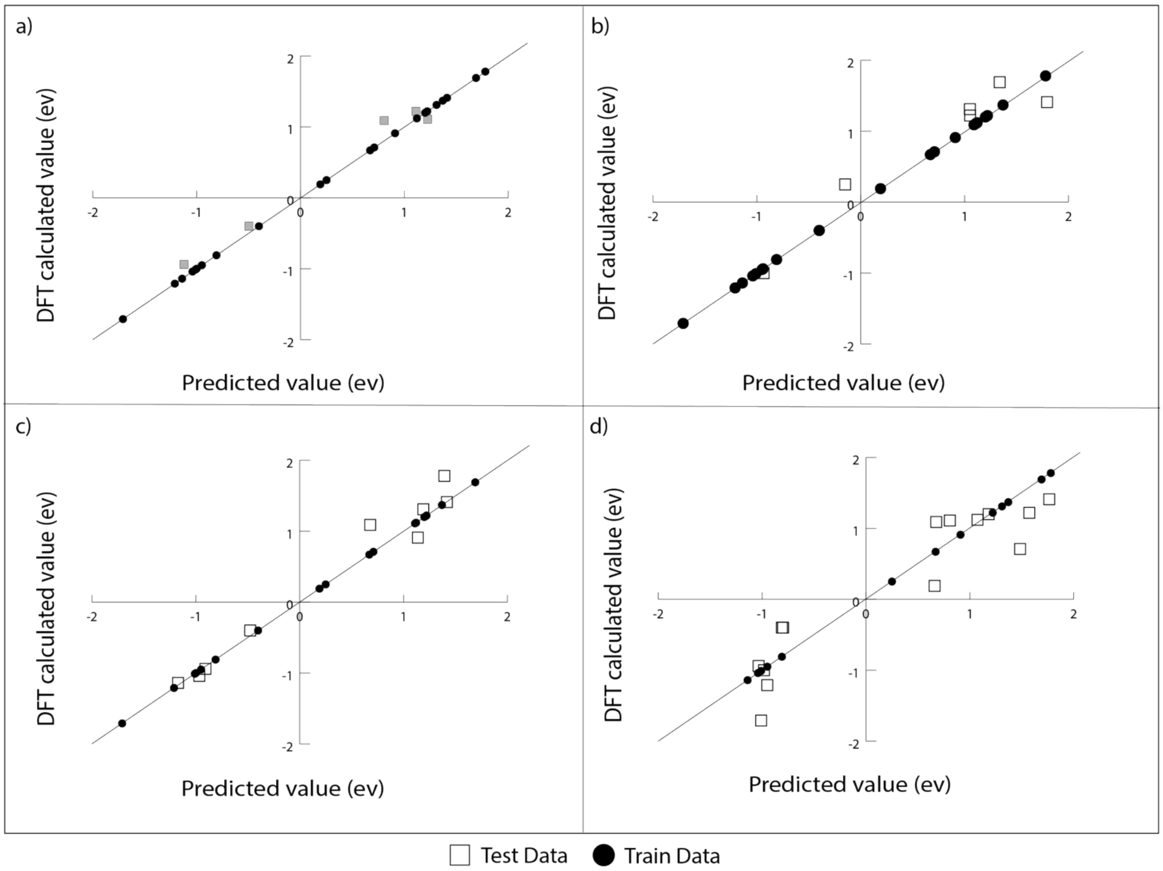
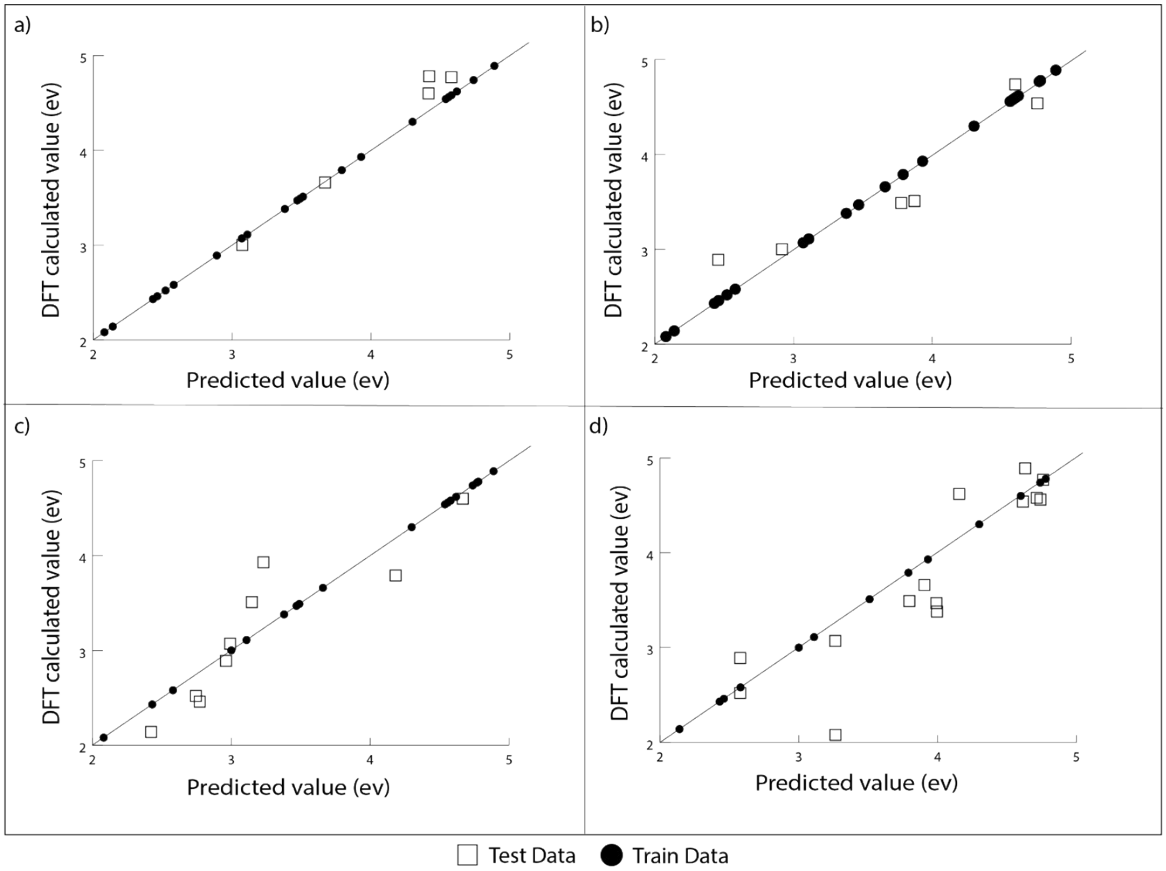


Figure SI-6. The deviation of DFT calculated oxygen binding energy with that predicted from the GBR model for Cu-based SAA for a) test/train ratio of 15/85 b) test/train ratio of 20/80 c) test/train ratio of 30/70 d) test/train ratio of 50/50



FigureSI-7. The deviation of DFT calculated carbon binding energy with that predicted from the GBR model for Cu-based SAA for a) test/train ratio of 15/85 b) test/train ratio of 20/80 c) test/train ratio of 30/70 d) test/train ratio of 50/50

Sample code for ML prediction of binding energies using GBR

```
#!/usr/bin/env python2
# -*- coding: utf-8 -*-
"""
Created on Tue Jun  4 13:20:37 2019

@author: shivamsaxena
"""

#Importing libraries to use.
import pandas as pd
import numpy as np
from math import sqrt
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from sklearn.ensemble import GradientBoostingRegressor


#Input path of the file that contains data
url = "(Location of csv file)"

# Reading the file using pandas and assigning X to independent variables and Y to dependent variable.
df = pd.read_csv(url, header=0)
Y=df.loc[:,'B.E.']
```

```
X= df.loc[:,'AN':'SE']

#Creating lists
errors_test=[]
errors_train=[]
feature_importances_array=np.zeros(12)

i=0
#The process is repeated 100 times to remove biasing due to data.
for i in range(0,100):
    #Dividing the data into test and train set with ratio of 0.2.
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2)
    #Initialising the GRB model
    clf=GradientBoostingRegressor(n_estimators=200, learning_rate=0.5, max_depth=4)
    #Fitting the train data to the model
    model=clf.fit(X_train, y_train)
    #Predict dependent variable of train set
    y_train_predicted=model.predict(X_train)
    #Predict dependent variable of test set
    y_test_predicted=model.predict(X_test)
    #Calculating the RMSE train error
    errors_train.append(sqrt(mean_squared_error(y_train, y_train_predicted)))
    #Calculating the RMSE test error
    errors_test.append(sqrt(mean_squared_error(y_test, y_test_predicted)))
    #Obtaining the relative feature importance of each independent variable for the model.
    feature_importance_step=model.feature_importances_
    feature_importances_array=np.add(feature_importances_array,feature_importance_step)
    i=i+1

#Printing the test error, train error and relative feature importance of variables
print "Train Error:" + " " + str(sum(errors_train)/float(len(errors_train)))
print "Test Error:" + " " + str(sum(errors_test)/float(len(errors_test)))
print "Feature importance values:" +" " + str(np.divide(feature_importances_array,100.0))
```